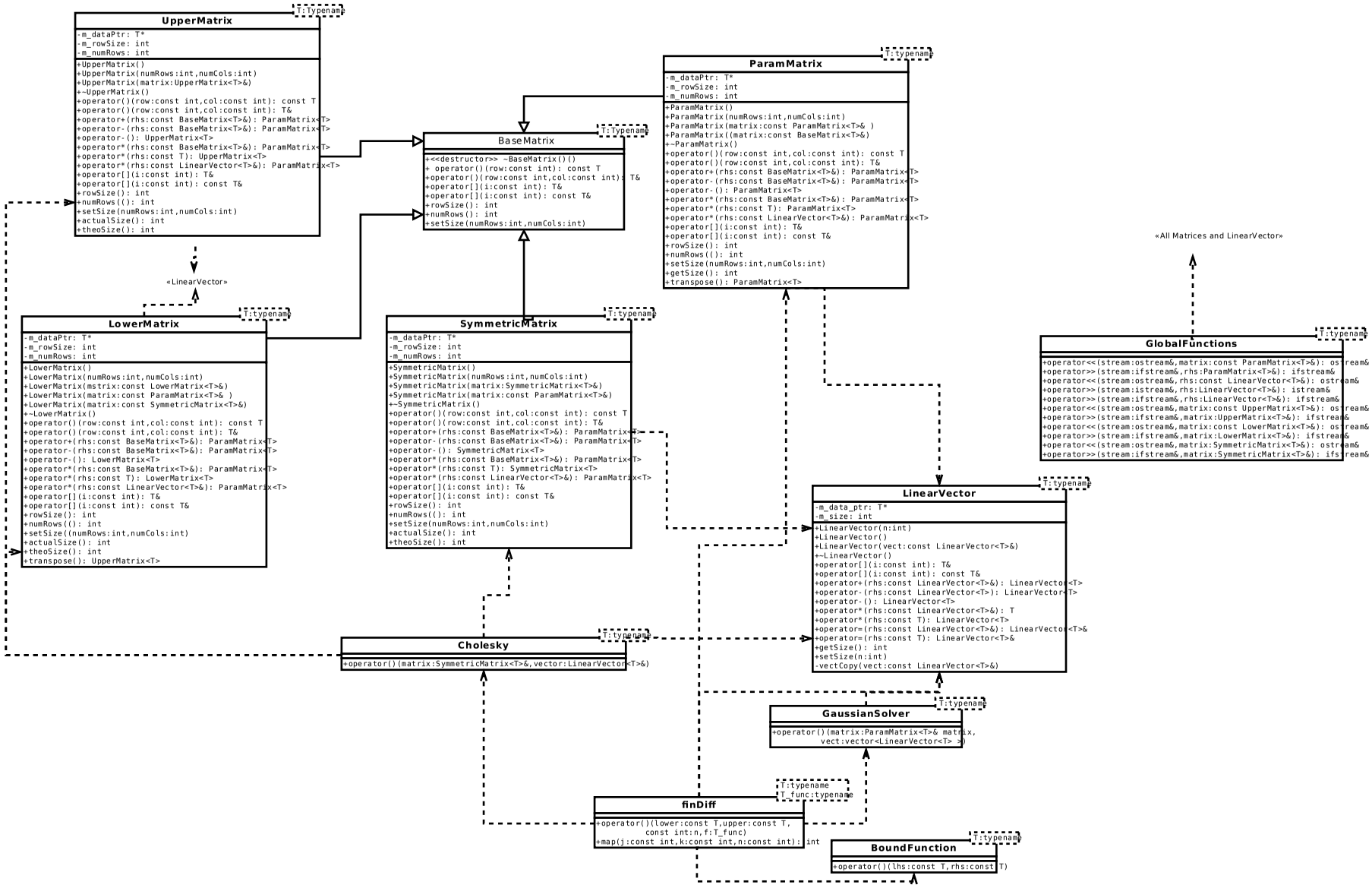
T:typename
UpperMatrix
-m_dataPtr: T*
-m_rowSize: int
-m_numRows: int
+UpperMatrix()
+UpperMatrix(numRows:int,numCols:int)
+UpperMatrix(matrix:UpperMatrix<T>&)
+~UpperMatrix()
+operator()(row:const int,col:const int): const T
+operator()(row:const int,col:const int): T&
+operator+(rhs:const BaseMatrix<T>&): ParamMatrix<T>
+operator-(rhs:const BaseMatrix<T>&): ParamMatrix<T>
+operator-(): UpperMatrix<T>
+operator*(rhs:const BaseMatrix<T>&): ParamMatrix<T>
+operator*(rhs:const T): UpperMatrix<T>
+operator*(rhs:const LinearVector<T>&): ParamMatrix<T>
+operator[](i:const int): T&
+operator[](i:const int): const T&
+rowSize(): int
+numRows(): int
+setSize(numRows:int,numCols:int)
+actualSize(): int
+theoSize(): int
T:Typename
BaseMatrix
+<<destructor>> ~BaseMatrix()()
+ operator()(row:const int): const T
+operator()(row:const int,col:const int): T&
+operator[](i:const int): T&
+operator[](i:const int): const T&
+rowSize(): int
+numRows(): int
+setSize(numRows:int,numCols:int)
T:typename
ParamMatrix
-m_dataPtr: T*
-m_rowSize: int
-m_numRows: int
+ParamMatrix()
+ParamMatrix(numRows:int,numCols:int)
+ParamMatrix(matrix:const ParamMatrix<T>&)
+ParamMatrix((matrix:const BaseMatrix<T>&)
+~ParamMatrix()
+operator()(row:const int,col:const int): const T
+operator()(row:const int,col:const int): T&
+operator+(rhs:const BaseMatrix<T>&): ParamMatrix<T>
+operator-(rhs:const BaseMatrix<T>&): ParamMatrix<T>
+operator-(): ParamMatrix<T>
+operator*(rhs:const BaseMatrix<T>&): ParamMatrix<T>
+operator*(rhs:const T): ParamMatrix<T>
+operator*(rhs:const LinearVector<T>&): ParamMatrix<T>
+operator[](i:const int): T&
+operator[](i:const int): const T&
+rowSize(): int
+numRows(): int
+setSize(numRows:int,numCols:int)
+getSize(): int
+transpose(): ParamMatrix<T>
«LinearVector»
«All Matrices and LinearVector»
T:typename
LowerMatrix
-m_dataPtr: T*
-m_rowSize: int
-m_numRows: int
+LowerMatrix()
+LowerMatrix(numRows:int,numCols:int)
+LowerMatrix(mstrix:const LowerMatrix<T>&)
+LowerMatrix(matrix:const ParamMatrix<T>&)
+LowerMatrix(matrix:const SymmetricMatrix<T>&)
+~LowerMatrix()
+operator()(row:const int,col:const int): const T
+operator()(row:const int,col:const int): T&
+operator+(rhs:const BaseMatrix<T>&): ParamMatrix<T>
+operator-(rhs:const BaseMatrix<T>&): ParamMatrix<T>
+operator-(): LowerMatrix<T>
+operator*(rhs:const BaseMatrix<T>&): ParamMatrix<T>
+operator*(rhs:const T): LowerMatrix<T>
+operator*(rhs:const LinearVector<T>&): ParamMatrix<T>
+operator[](i:const int): T&
+operator[](i:const int): const T&
+rowSize(): int
+numRows(): int
+setSize((numRows:int,numCols:int)
+actualSize(): int
+theoSize(): int
+transpose(): UpperMatrix<T>
T:typename
SymmetricMatrix
-m_dataPtr: T*
-m_rowSize: int
-m_numRows: int
+SymmetricMatrix()
+SymmetricMatrix(numRows:int,numCols:int)
+SymmetricMatrix(matrix:SymmetricMatrix<T>&)
+SymmetricMatrix(matrix:const ParamMatrix<T>&)
+~SymmetricMatrix()
+operator()(row:const int,col:const int): const T
+operator()(row:const int,col:const int): T&
+operator+(rhs:const BaseMatrix<T>&): ParamMatrix<T>
+operator-(rhs:const BaseMatrix<T>&): ParamMatrix<T>
+operator-(): SymmetricMatrix<T>
+operator*(rhs:const BaseMatrix<T>&): ParamMatrix<T>
+operator*(rhs:const T): SymmetricMatrix<T>
+operator*(rhs:const LinearVector<T>&): ParamMatrix<T>
+operator[](i:const int): T&
+operator[](i:const int): const T&
+rowSize(): int
+numRows(): int
+setSize(numRows:int,numCols:int)
+actualSize(): int
+theoSize(): int
T:typename
GlobalFunctions
+operator<<(stream:ostream&,matrix:const ParamMatrix<T>&): ostream&
+operator>>(stream:ifstream&,rhs:ParamMatrix<T>&): ifstream&
+operator<<(stream:ostream&,rhs:const LinearVector<T>&): ostream&
+operator>>(stream:istream&,rhs:LinearVector<T>&): istream&
+operator>>(stream:ifstream&,rhs:LinearVector<T>&): ifstream&
+operator<<(stream:ostream&,matrix:const UpperMatrix<T>&): ostream&
+operator>>(stream:ifstream&,matrix:UpperMatrix<T>&): ifstream&
+operator<<(stream:ostream&,matrix:const LowerMatrix<T>&): ostream&
+operator>>(stream:ifstream&,matrix:LowerMatrix<T>&): ifstream&
+operator<<(stream:ostream&,matrix:SymmetricMatrix<T>&): ostream&
+operator>>(stream:ifstream&,matrix:SymmetricMatrix<T>&): ifstream&
T:typename
LinearVector
-m_data_ptr: T*
-m_size: int
+LinearVector(n:int)
+LinearVector()
+LinearVector(vect:const LinearVector<T>&)
+~LinearVector()
+operator[](i:const int): T&
+operator[](i:const int): const T&
+operator+(rhs:const LinearVector<T>&): LinearVector<T>
+operator-(rhs:const LinearVector<T>): LinearVector<T>
+operator-(): LinearVector<T>
+operator*(rhs:const LinearVector<T>&): T
+operator*(rhs:const T): LinearVector<T>
+operator=(rhs:const LinearVector<T>&): LinearVector<T>&
+operator=(rhs:const T): LinearVector<T>&
+getSize(): int
+setSize(n:int)
-vectCopy(vect:const LinearVector<T>&)
T:typename
Cholesky
+operator()(matrix:SymmetricMatrix<T>&,vector:LinearVector<T>&)
T:typename
GaussianSolver
+operator()(matrix:ParamMatrix<T>& matrix,
vect:vector<LinearVector<T> >)
T:typename
T_func:typename
finDiff
+operator()(lower:const T,upper:const T,
const int:n,f:T_func)
+map(j:const int,k:const int,n:const int): int
T:typename
BoundFunction
+operator()(lhs:const T,rhs:const T)